Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (currently amended): A method for mapping procedural code to object-oriented classes, comprising:

starting a graphical user interface ("GUI") in a procedural programming language process space, wherein a user enters a command through the GUI;

initializing a mapping layer in an object-oriented programming language process space, wherein the mapping layer comprises entry-points that have corresponding algorithms that invoke object-oriented class instantiation methods and/or remote method invocations ("RMIs");

executing a GUI callback in response to the command, wherein the GUI callback comprises procedural code and wherein executing a GUI callback in response to the command comprises:

invoking one of the entry-points, wherein the object-oriented programming language is Java, the mapping layer is accessed through a Java Native Interface ("JNI") and invoking one of the entry-points comprises:

invoking a JNI application programming interface ("API") call, wherein the JNI API call invokes one of the entry-points; ~~and~~

the mapping layer executing an algorithm corresponding to the invoked entry-point; and

wherein the mapping layer is proxied in the procedural programming language process space by a proxy object that includes proxy object methods corresponding to the entry-points, and executing a GUI callback further comprises:

invoking one of the proxy object methods, wherein the invoked proxy object method performs the invoking one of the entry-points.

Claim 2 (original): The method of claim 1, wherein executing the algorithm comprises invoking a class instantiation method.

Claim 3 (original): The method of claim 1, wherein executing the algorithm comprises invoking an RMI.

Claim 4 (original): The method of claim 1, wherein the procedural programming language is C++.

Claims 5-7 (canceled).

Claim 8 (original): The method of claim 1, wherein the entry-points are methods of the mapping layer.

Claim 9 (original): The method of claim 1, further comprising returning data to the procedural programming language process space.

Claim 10 (currently amended): A computer readable medium containing instructions for mapping procedural code to object-oriented classes, by:

starting a graphical user interface ("GUI") in a procedural programming language process space, wherein a user enters a command through the GUI;

initializing a mapping layer in an object-oriented programming language process space, wherein the mapping layer comprises entry-points that have corresponding algorithms that invoke object-oriented class instantiation methods and/or remote method invocations ("RMIs");

executing a GUI callback in response to the command, wherein the GUI callback comprises procedural code and wherein executing a GUI callback in response to the command comprises:

invoking one of the entry-points, wherein the object-oriented programming language is Java, the mapping layer is accessed through a Java Native Interface ("JNI") and invoking one of the entry-points comprises:

invoking a JNI application programming interface ("API") call, wherein the JNI API call invokes one of the entry-points; ~~and~~

the mapping layer executing an algorithm corresponding to the invoked entry-point; and

wherein the mapping layer is proxied in the procedural programming language process space by a proxy object that includes proxy object methods corresponding to the entry-points, and executing a GUI callback further comprises:

invoking one of the proxy object methods, wherein the invoked proxy object method performs the invoking one of the entry-points.

Claim 11 (original): The computer readable medium of claim 10, wherein executing the algorithm comprises invoking a class instantiation method.

Claim 12 (original): The computer readable medium of claim 10, wherein executing the algorithm comprises invoking an RMI.

Claim 13 (original): The computer readable medium of claim 10, wherein the procedural programming language is C++.

Claims 14-15 (canceled).

Claim 16 (currently amended): A computer system that enables the mapping of procedural code to object-oriented classes, comprising:

a memory;

a processor that runs an application, wherein the application generates:

a graphical user interface ("GUI") in a procedural programming language process space, wherein users enter commands through the GUI; and~~,~~

a mapping layer in an object-oriented programming language process space, wherein the mapping layer comprises entry-points that have corresponding algorithms that invoke object-oriented class instantiation methods and/or remote method invocations ("RMIs"); and

wherein the application includes instructions for:

executing a GUI callback in response to the command, wherein the GUI callback comprises procedural code and wherein executing a GUI callback in response to the command comprises:

invoking one of the entry-points, wherein the object-oriented programming language is Java, the mapping layer is accessed through a Java Native Interface ("JNI") and invoking one of the entry-points comprises:

invoking a JNI application programming interface ("API") call, wherein the JNI API call invokes one of the entry-points; ~~and~~

the mapping layer executing an algorithm corresponding to the invoked entry-point; and wherein the mapping layer is proxied in the procedural programming language process space by a proxy object that includes proxy object methods corresponding to the entry-points, and executing a GUI callback further comprises:

invoking one of the proxy object methods, wherein the invoked proxy object method performs the invoking one of the entry-points.

Claim 17 (original): The computer system of claim 16, wherein the GUI executes callback code in response to an entered command and the executed callback code invokes one of the mapping layer entry-points.

Claim 18 (original): The computer system of claim 16, wherein the entry-points are mapping layer methods that are accessed from the procedural programming language process space though application programming interface ("API") calls.

Claim 19 (original): The computer system of claim 16, wherein the procedural programming language is C++.

Claim 20 (currently amended): The computer system of claim 16, wherein ~~the object-oriented programming language is Java and~~ the object-oriented programming language process space is a Java Virtual Machine.